

Ex. No. 5	JAVASCRIPT – DOM ELEMENTS AND EVENTS
Date of Exercise	09.08.2023

Aim

To create a javascript DOM elements and events using javascript.

Description

- A Document object represents the HTML document that is displayed in that window.
- Document object has various properties that refer to other objects which allow access to and modification of document content.
- The way a document content is accessed and modified is called the Document Object Model, or DOM.
- `var myElement = document.getElementById("intro");`
- The JavaScript syntax defines two types of values Fixed values are called Literals, Variable values are called Variables.
- In a programming language, variables are used to store data values, JavaScript uses the keywords `var`, `let` and `const` to declare variables.
- An equal sign is used to assign values to variables, All JavaScript variables must be identified with unique names.
- These unique names are called identifiers. A JavaScript function is defined with the function keyword, followed by a name, followed by parentheses `()`.
- Function names can contain letters, digits, underscores, and dollar signs (same rules as variables). When JavaScript reaches a return statement, the function will stop executing.
- If the function was invoked from a statement, JavaScript will "return" to execute the code after the invoking statement. Functions often compute a return value.
- JavaScript functions are used to perform operations. We can call Javascript function many times to reuse the code. In JavaScript the purpose of function constructor is to create a new function object.

Program

- 1) Design a webpage using HTML, CSS and JavaScript as per the following design and perform the Cake order calculation using JavaScript. [use onclick event]

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Cake Order</title>
<style>
  body {
    font-family: Arial, sans-serif;
  }
  #cake-options {
    display: flex;
    flex-direction: column;
    gap: 10px;
    margin-bottom: 20px;
  }
  .cake-option {
    display: flex;
    align-items: center;
    gap: 10px;
  }
  .grid-container {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
    gap: 20px;
  }
  .item {
    text-align: center;
  }
  .item img {
    max-width: 100%;
    height: auto;
  }
```

```

        .center-content {
            display: flex;
            justify-content: center;
            align-items: center;
        }
    </style>
</head>
<body>
    <div style="background-color: red;text-align: center;">
        <h1>Cake shop</h1>
    </div>

    <div class="grid-container">
        <div class="item">
            <br>
            <label>Brown Cake-Rs. 200</label>
        </div>
        <div class="item">
            <br>
            <label>Milk Cake-Rs. 250</label>
        </div>
        <div class="item">
            
            <br><label>Black Forest Cake-Rs. 350</label>
        </div>
        <div class="item">
            <br>
            <label>Color Cake-Rs. 300</label>
        </div>

    </div>

    <div class="center-content">
        <h1>Order Cake</h1></div>
    <div class="center-content">
        <div id="cake-options">
            <div class="cake-option">
                <label for="chocolate">Color Cake - Rs 300</label>

```

```
<input type="number" id="chocolate" min="0" value="0"
onchange="calculateTotal()">
</div>
<div class="cake-option">
  <label for="vanilla">Milk Cake - Rs 250</label>
  <input type="number" id="vanilla" min="0" value="0" onchange="calculateTotal()">
</div>
<div class="cake-option">
  <label for="strawberry">Black Forest Cake - Rs 350</label>
  <input type="number" id="strawberry" min="0" value="0"
onchange="calculateTotal()">
</div>
<div class="cake-option">
  <label for="lemon">Brown Cake - Rs 200</label>
  <input type="number" id="lemon" min="0" value="0" onchange="calculateTotal()">
</div>
</div>
<div>
<p style="text-align: center;">Total Cost: $<span id="total">0</span></p>
<script>
function calculateTotal() {
  const chocolateQuantity = parseInt(document.getElementById('chocolate').value);
  const vanillaQuantity = parseInt(document.getElementById('vanilla').value);
  const strawberryQuantity = parseInt(document.getElementById('strawberry').value);
  const lemonQuantity = parseInt(document.getElementById('lemon').value);
  const chocolatePrice = 300;
  const vanillaPrice = 200;
  const strawberryPrice = 350;
  const lemonPrice = 200;
  const totalCost = (chocolateQuantity * chocolatePrice) +
    (vanillaQuantity * vanillaPrice) +
    (strawberryQuantity * strawberryPrice) +
    (lemonQuantity * lemonPrice);

  document.getElementById('total').textContent = totalCost;
}
</script>
</body>
</html>
```

- 2) Create a web application to develop the following Bank loan EMI calculator using client side Java Script. The formula for calculating the EMI is given below. Where, P is Loan Amount, R is a rate of interest and N is a tenure in months. [convert Year as months] [Use onkeyup event]

```
<!DOCTYPE html>
<html>
<head>
  <title>Bank Loan EMI Calculator</title>
  <style>
    body {
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
      margin: 0;
      background-color: #f0f0f0;
    }

    .calculator {
      text-align: left;
      padding: 20px;
      border-radius: 10px;
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);
      width: 300px;
    }

    label {
      display: block;
      margin-top: 10px;
    }

    .result-heading {
      background-color: #3498db;
      color: #fff;
      padding: 10px;
      border-radius: 5px;
      margin-top: 10px;
      font-weight: bold;
    }
```

```
    }

    .result-bar {
        background-color: #e0e0e0;
        padding: 10px;
        border-radius: 5px;
        margin-top: 5px;
        font-weight: bold;
    }
</style>
<script>
    function formatCurrency(amount) {
        return '₹' + amount.toFixed(2);
    }

    function calculateEMI() {
        var loanAmount =
parseFloat(document.getElementById("loanAmountInput").value);
        var annualInterestRate =
parseFloat(document.getElementById("annualInterestRate").value) / 100;
        var monthlyInterestRate = annualInterestRate / 12;
        var tenureInYears =
parseFloat(document.getElementById("tenureInYears").value);
        var tenureInMonths = tenureInYears * 12;

        var emi = (loanAmount * monthlyInterestRate * Math.pow(1 +
monthlyInterestRate, tenureInMonths)) / (Math.pow(1 + monthlyInterestRate,
tenureInMonths) - 1);
        var totalRepayment = emi * tenureInMonths;
        var totalInterest = totalRepayment - loanAmount;

        document.getElementById("loanAmount").innerHTML =
formatCurrency(loanAmount);
        document.getElementById("emiResult").innerHTML = formatCurrency(emi);
        document.getElementById("totalRepaymentResult").innerHTML =
formatCurrency(totalRepayment);
        document.getElementById("totalInterestResult").innerHTML =
formatCurrency(totalInterest);
    }
</script>
```

```
</head>
<body>
  <div class="calculator">
    <h1>Bank Loan EMI Calculator</h1>

    <label for="loanAmountInput">Loan Amount:</label>
    <input type="number" id="loanAmountInput" placeholder="Loan Amount"
onkeyup="calculateEMI()">

    <label for="annualInterestRate">Annual Interest Rate (%):</label>
    <input type="number" id="annualInterestRate" placeholder="Annual Interest Rate"
onkeyup="calculateEMI()">

    <label for="tenureInYears">Tenure in Years:</label>
    <input type="number" id="tenureInYears" placeholder="Tenure in Years"
onkeyup="calculateEMI()">

    <div class="result-heading">Loan Amount</div>
    <div class="result-bar" id="loanAmount"></div>


    <div class="result-heading">EMI</div>
    <div class="result-bar" id="emiResult"></div>


    <div class="result-heading">Total Repayment</div>
    <div class="result-bar" id="totalRepaymentResult"></div>


    <div class="result-heading">Total Interest</div>
    <div class="result-bar" id="totalInterestResult"></div>
  </div>
</body>
</html>
```


Output

Cake shop


Brown Cake-Rs. 200


Milk Cake-Rs. 250


Black Forest Cake-Rs. 350


Color Cake-Rs. 300

Order Cake

Color Cake - Rs 300

Milk Cake - Rs 250

Black Forest Cake - Rs 350

Brown Cake - Rs 200

Total Cost: \$0

Bank Loan EMI Calculator

Loan Amount:

Annual Interest Rate (%):

Tenure in Years:

Loan Amount

EMI

Total Repayment

Total Interest

Result

The program is executed successfully and the program output is displayed in the web browser.